UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/943,599	08/30/2001	Gary L. Swoboda	TI-30481	2479

23494 7590 12/05/2006

TEXAS INSTRUMENTS INCORPORATED
P O BOX 655474, M/S 3999
DALLAS, TX 75265

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 12/05/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/943,599
Filing Date: August 30, 2001
Appellant(s): SWOBODA ET AL.

Robert D. Marshall, Jr.
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 10th March 2006 appealing from the Office action mailed 4th January 2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is incorrect. A correct statement of the status of the claims is as follows:

This appeal involves claims 1, 4, 5, 13, 15, 16, 23, 24 and 27-30.

Claims 2-3, 6-12, 14, 17-22 and 25-26 been canceled.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

No evidence is relied upon by the examiner in the rejection of the claims under appeal.

(9) Prior Art of Record

U.S. Patent No 6,009,270 issued to Daniel Mann.

U.S. Patent No. 5,764,885 issued to Richard L. Sites et al

U.S. Patent No. 6,732,307 issued to David A. Edwards

(9) Grounds of Rejection

1. Claim 1-10 and 12 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No 6,009,270 issued to Daniel Mann (Mann '270 hereafter).

Regarding Claim 1

Mann '270 teaches:

> "A method of providing data processor emulation information, comprising:
> providing a program counter trace stream of program counter values used by a data processor;"

as a trace stream (Mann '270: Col.14, Lines 48-50) containing address information for reconstructing instruction execution flow (Mann '270: Col. 2, Lines 60-64). It is known in the art that program counter contains op-code and address information for the operands for each instruction. A stream of program counter will provide an instruction flow in a program counter. Hence the stream taught by Mann '270 is equivalent to program counter trace stream.

Mann '270 also teaches:

> "Inserting a synchronization marker into the program counter trace stream; and"

as insertion of a synchronizing address information (current program address) to synchronize trace based on the state of synchronization register (TSYNCH) (Mann '270: Col.16, Lines 3-5; Col.16, Lines 25-29).

Mann '270 also teaches:

> "providing trace information indicative of each data processing operation performed by the data processor, including identifying a program counter value that corresponds to the data processing operation, said identifying step including expressing said corresponding program counter value as an offset which indicates a number of program counter values in the program counter trace stream by which said corresponding program counter value is offset from said synchronization marker in said program counter trace stream."

Mann '270 teaches providing trace information indicative of the instruction execution flow, which includes all processing operations (Col.2 Lines 60 – Col.3 Lines 30; Especially Col.2 Lines 21-23). As one stream trace containing program execution information including synchronization markers (Mann '270: Abstract Lines1-4) performing data processing and I/O operations (Mann '270: Col.10, Lines 9-11). Mann further teaches that branching causes disruption in the flow and non-data dependent branching (where new address is not dependent on address in data registers) (Mann '270: Col.13, Lines 56-62) can be represented in a form of an offset indicating whether the branch was taken or not. This offset information is presented in form of a trace entry (Mann '270: Col.14, Lines 7-16) from which new target value can be reconstructed (Mann '270: Col. 15 Lines 8-13, 53-56).

Regarding Claim 4

Mann '270 clearly teaches that conditional branching statements (TCODE=1) are relevant only after the some base address (complete PC address) is provided by other TCODE instructions (Col.15 Lines 42-52; Col.16, Lines 13-15). Further, in the conditional branching statement a new program counter is loaded based on if the branch is taken, each branch will lead to new value in PC, hence detecting a new branch will inherently mean detecting occurrences program counter load.

Further, Mann '270 teaches identifying step to include counting detected occurrences of program counter load as follows. Each bit in the trace command (TCODE=1) represents a single program counter load; hence the number of bits used (up to 15) in the trace command will indicate the number of program counter

loads. The trace command besides storing the decision for branch also acts as a counter of how many program counter loads have taken place (Table 6; Col.14 Lines 7-36).

Regarding Claim 5

Claim 5 inherits all its limitation from claim 4, thus rejection presented above also pertains to claim 5 above. Further, claim 5 discloses maintaining a running count of number of program counter loads. Mann '270 teaches, as disclosed above, a running count as number of bits in the TCODE=1 statement, that have occurred after the insertion of the synchronization marker (all TCODE statements except TCODE=1) (Table 6; Col.15 Lines 42-52).

2. Claims 13,15 & 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,764,885 issued to Richard L. Sites et al (Sites '885 hereafter) in view of U.S. Patent No. 6,009,270 issued to Daniel Mann (Mann '270 hereafter).

Regarding Claim 13

Sites '885 teaches:

> An apparatus for providing data processor emulation information, comprising:
> first and second inputs for coupling to a data processor;

as an apparatus containing a data processor (Sites '885: Figure 1C, Element 139) having two inputs each representing data stream from instruction cache (Sites '885: Figure 1C, Element 137) and data cache (Sites '885: Figure 1C, Element 138).

Sites '885 also teaches:

> a trace stream generator coupled to said first input for providing a program counter trace stream of program counter values used by the data processor, said trace stream generator operable for inserting a synchronization marker into the program counter trace stream;

as a trace stream generator (Sites '885: Figure 1C, Element 150, I Trace Logic) coupled to first input providing a program counter trace stream (Sites '885: Figure 1C, Element 101) through the aforementioned data processor. I Trace Logic described by Sites '885 also inserts the synchronization markers in the trace stream entries (Sites '885: Figure 2, Elements 210,220& 230) as time indices (Sites '885: Figure 2, Element 202; Col.5, Lines 61-67).

Sites '885 also teaches:

> and a trace apparatus coupled to said second input for providing trace information indicative of a data processing operation performed by the data processor, including a program counter identifier for identifying a program counter value that corresponds to said data processing operation, said program counter identifier operable for expressing said corresponding program counter value as an offset which indicates a number of program counter values in the program

counter trace stream by which said corresponding program counter value is offset from said synchronization marker in said program counter trace stream.

As a trace apparatus mapped to D Trace Logic (Sites '885: Figure 1C, Element 160) connected to second input indicative of data processing operation, concerning data, coming from D Cache (Sites '885: Figure 1C, Element 135). Sites also teaches how function of the program counter identifier can implemented to create program counter values for direct program counter loads (Sites '885: Col.9, Lines 60-67) and conditional branching (Sites '885: Col.9, Lines 1-14). Further, Sites '885 teaches creating a synchronization marker in the second stream, wherein the synchronization marker is in the form of time stamp (Sites '885: Col.5, Lines 61-67; Figure 2, Element 202) in a data processing entry (Sites '885: Figure 2, Element 270).

Sites '885 do not teach creating a program counter offset based on the synchronization marker in the same stream.

Mann '270 above teaches using a synchronization marker (Mann '270: Col.16, Lines 3-5; Col.16, Lines 25-29). Further, This offset information is presented in form of a trace entry (Mann '270: Col.14, Lines 7-16) from which new target value can be reconstructed (Mann '270: Col. 15 Lines 8-13, 53-56).

It would have been obvious to one (e.g. a designer) of ordinary skill in the art at the time the invention was made to take Mann '270's teachings and apply them to Sites '885 design of using separate streams for program counter and data processing operations. The motivation would be that such synchronization would provide address information frequently enough for reconstructing instruction

execution flow with offset in second stream, making a tighter coupling between the two streams. Time indices used in the Sites '885 reference provide good idea what a target PC value was currently used, looking at the stream history (Sites '885: Figure 2) but may not be provided frequently enough. Mann's concept of counter based synchronization overcomes such a loss of reference to tie two streams together.

Regarding Claim 15

The new claim 15 amends in the limitations of claim 14. Mann '270 clearly teaches that conditional branching statements (TCODE=1) are relevant only after the some base address (complete PC address) is provided by other TCODE instructions (Col.15 Lines 42-52; Col.16, Lines 13-15). Further, in the conditional branching statement a new program counter is loaded based on if the branch is taken, each branch will lead to new value in PC, hence detecting a new branch will inherently mean detecting occurrences program counter load.

Further, Mann '270 teaches identifying step to include counting detected occurrences of program counter load as follows. Each bit in the trace command (TCODE=1) represents a single program counter load; hence the number of bits used (up to 15) in the trace command will indicate the number of program counter loads. The trace command besides storing the decision for branch also acts as a counter of how many program counter loads have taken place (Table 6; Col.14 Lines 7-36).

Regarding Claim 16

Claim 16 inherits all its limitation from claim 15, thus rejection presented above also pertains to claim 15 above. Further, claim 15 discloses maintaining a running count of number of program counter loads. Mann '270 teaches, as disclosed above, a running count as number of bits in the TCODE=1 statement, that have occurred after the insertion of the synchronization marker (all TCODE statements except TCODE=1) (Table 6; Col.15 Lines 42-52).

3. **Claims 23-24 and 27-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,764,885 issued to Richard L. Sites et al (Sites '885 hereafter) in view of U.S. Patent No. 6,009,270 issued to Daniel Mann (Mann '270 hereafter), further in view of U.S. Patent No. 6,732,307 issued to David A. Edwards (Edwards '307 hereafter).**

Regarding Claim 23

Sites '885 and Mann '270 teachings' are mentioned above in rejection for claim 13. Sites '885 and Mann '270 teaches an apparatus with same limitations as claim 23 above.

Sites '885 and Mann '270 do no disclose an integrated circuit with the limitations disclosed in claim 13.

Edwards '307 teaches an integrated circuit design which extract and export the trace information from a data processor (Edwards '307: Figure 1), hence performing the same function and being is analogous art to Sites '885 and Mann '270.

Motivation to combine Sites '885 and Mann '270 is provided in the claim 13 rejection above.

It would have been obvious to one (e.g. a designer) of ordinary skill in the art at the time the invention was made to take teachings of Edwards '307 and apply them to Sites '885 and Mann '270. The motivation would have been to reduce bandwidth for off-chip trace transmission to an external system by providing on-chip trace handling and compression (Edwards '307: Col.1, Lines 13; Col.2 Lines 41-44). Further, arguments provided below to clarify appellant's arguments also apply here.

Regarding Claim 24

Sites '885 and Mann '270 teachings are disclosed in rejection for claim13. Sites '885 and Mann '270 teach an apparatus with same limitations as claim 24 above. Mann '270 also teaches an emulation controller to load the instructions that need to be executed on the target data processor (Mann '270: Col.4, Lines 10-13 & Figure 1; Col.4, Lines 55-57 & Figure 2).

Sites '885 and Mann '270 do no disclose an integrated circuit system with the limitations disclosed above.

Edwards '307 teaches an integrated circuit system (Edwards '307: Figure 1) containing a debug circuit (Edwards '307: Figure 2, Element 103 - emulation controller) as well, performing the same function as disclosed by Sites '885 and Mann '270.

It would have been obvious to one (e.g. a designer) of ordinary skill in the art at the time the invention was made to take teachings of Edwards '307 and apply them to Sites '885 and Mann '270. The motivation would have been to reduce bandwidth for off-chip trace transmission to an external system by providing on-chip trace handling and compression (Edwards '307: Col.1, Lines 13; Col.2 Lines 41-44). Further motivation comes from the fact that system-on-chip (SOC) concept is well known in the art with its benefits to reduce intersystem delays and speed up communication among systems on the chip. In our present invention debug circuit system interfaces with the data processor and is a part of SOC (Edwards '307: Figure 1).

Regarding Claims 27 & 29

Claims 27 and 29 disclose substantially similar limitation for an integrated circuit and a data processing system. Mann '270 clearly teaches that conditional branching statements (TCODE=1) are relevant only after the some base address (complete PC address) is provided by other TCODE instructions (Col.15 Lines 42-52; Col.16, Lines 13-15). Further, in the conditional branching statement a new program counter is loaded based on if the branch is taken, each branch will lead to new value in PC, hence detecting a new branch will inherently mean detecting occurrences program counter load.

Further, Mann '270 teaches identifying step to include counting detected occurrences of program counter load as follows. Each bit in the trace command (TCODE=1) represents a single program counter load; hence the number of bits used (up to 15) in the trace command will indicate the number of program counter loads. The trace command besides storing the decision for branch also acts as a counter of how many program counter loads have taken place (Table 6; Col.14 Lines 7-36).

Regarding Claims 28 & 30

Claims 28 & 30 inherit their limitation from claims 27 and 29 respectively, thus rejection presented above also applies here. Claim 27 & 29 discloses maintaining a running count of number of program counter loads. Mann '270 teaches, as disclosed above, a running count as number of bits in the TCODE=1 statement, that have

occurred after the insertion of the synchronization marker (all TCODE statements except TCODE=1) (Table 6; Col.15 Lines 42-52).

(10) Response to Argument

Regarding Claim 1

It is noted that appellant's arguments regarding prior art have essentially only recited the prior art teaching followed by a recitation of the claims without pointing out the patentable distinction between the claimed invention and the prior art, or have been addressed to limitations that have not been specifically claimed.

In arguments relating to the prior art, appellants have taken the incorrect mapping of cited teachings and applied then against the claim limitations. Further, the appellant is limiting scope of the teachings of the prior art to only the cited portions. The portions of the prior art cited teach the limitation, however the inherent or salient features, as would be known to a skilled artisan in the art of trace processing, were not specifically cited.

For example appellant has argued that cited Mann portion (Col.13 Lines 56-62) does not teach the limitation:

> "**offset** which indicates a **number of program counter values** in the program counter trace stream by which said corresponding program counter value is offset from said **synchronization marker**..."

Mann teaches **synchronization marker** as TCODE entries in the trace stream, where all TCODE (except TCODE =0001) are synchronization events (*Mann: Col.16 Lines 13-16; Table 6*).

Mann Col.16 Lines 13-16:

> *The processor determines whether each trace record includes address information by, e.g., assuming all TCODES except for TCODE=1 are synchronizing events providing address information.*

Mann Table 6:

TABLE 6

TCODE #	TCODE Type	TDATA
0000	Missed Trace	Not Valid
0001	Conditional Branch	Contains Branch Sequence
0010	Branch Target	Contains Branch Target Address
0011	Previous Segment Base	Contains Previous Segment Base Address and Attributes
0100	Current Segment Base	Contains Current Segment Base Address and Attributes
0101	Interrupt	Contains Vector Number of Exception or Interrupt
0110	Trace Synchronization	Contains Address of Most Recently Executed Instruction
0111	Multiple Trace	Contains 2nd or 3rd Record of Entry With Multiple Records
1000	Trace Stop	Contains Instruction Address Where Trace Capture Was Stopped
1001	User Trace	Contains User Specified Trace Data
1010	Performance Profile	Contains Performance Profiling Data

TCODE=0001 provides the subsequent **offset information,** as 15 one bit results for subsequent branching outcomes, offset from the base address. The limitation "number of program counter values by which corresponding program counter is offset from the" base address (provided by synchronization marker), is taught as **cumulative offset value from the base address based on the branching result held in the TCODE 0001** where bits indicating if branch was taken, are used to calculate the subsequent program counter addresses (Mann: Col.14 – provided in TCODE 0010,0111 before the TCODE 0001 branching trace; Fig 6A,B; *Col.15 Lines 42-52*). Each conditional branch will decide the subsequent **number of counter values** to be added to the base (program counter) address. Hence the examiner asserts that the offset (as indicated by conditional branch bit of TCODE 0001)

indicating the program counter value in a trace stream from the synchronization marker (TCODE - See Mann Table 6 specifically TCODE = 0110, which provides the base address of the program counter) is taught by Mann.

Appellant again argues, by incorrectly mapping, cited teaching of Mann (Col.15 Lines 8-13, 53-56) as not teaching a synchronization marker and stating there is no teaching in Mann that " synchronization marker is the same as segment base address". As can be seen clearly from Mann Table 6 TCODE 0011 and Col.16 Lines 13-16 (See pg. 16 above), above statement by appellant is incorrect.

Mann teaches in Col.14 Lines 7-16 how offset is presented in each bit of TCODE=1:

> *"FIG. 6A illustrates an exemplary format for reporting conditional branch events. In the disclosed embodiment of the invention, the outcome of up to 15 branch events can be grouped into a single trace entry. The 16-bit TDATA field (or 'BFIELD') contains 1-bit branch outcome trace entries, and is labeled as a TCODE=0001 entry. The TDATA field is initially cleared except for the left most bit, which is set to 1. As each new conditional branch is encountered, a new one bit entry is added on the left and any other entries are shifted to the right by one bit."*

For clarification the salient feature, now specifically mentioned below, is the need for a base "program address" (program counter) to be able to examine the trace stream which is an offset from the base program counter address, as taught by Mann in

Col.15 Lines 42-52:

> *When executing typical software on a processor-based device 102 according to the disclosed embodiment of the invention, **few trace entries contain address values**. **Most entries are of the TCODE=0001 format**, in which a single bit indicates the result of a conditional operation. When examining a trace stream, however, **data can only be studied in relation to a known program address**. For example, starting with the oldest entry in the trace cache 200, all entries until an address entry are of little use. **Algorithm synchronization typically begins from a trace entry providing a target address.***

Specifically, TCODE = 0010, 0110 and 0111 provide base address (**complete program counter**) information (Table 6 & Col.14 Lines 17-36).

Mann: Col. 14 Lines 17-36:

Using a 128 entry trace cache 200 allows 320 bytes of information to be stored. Assuming a branch frequency of one branch every six instructions, the disclosed trace cache 200 therefore provides an effective trace record of 1,536 instructions. This estimate does not take into account the occurrence of call, jump and return instructions.

In the disclosed embodiment of the invention, the trace control logic 218 monitors instruction execution via processor interface logic 202. When a branch target address must be reported, information contained within a current conditional branch TDATA field is marked as complete by the trace control logic 218, even if 15 entries have not accumulated. As shown in FIG. 6B, the target address (in a processor-based device 102 using 32-bit addressing) is then recorded in a trace entry pair, with the first entry (TCODE=0010) providing the high 16-bits of the target address and the second entry (TCODE=0111) providing the low 16-bits of the target address. When a branch target address is provided for a conditional jump instruction, no 1-bit branch outcome trace entry appears for the reported branch.

Appellant's argues using Mann Col.3 Lines 5-7 that

"However, Mann fails to teach that the trace address values following such a synchronization marker are offset from the synchronization marker value."

Mann: Col.3 Lines 5-7 states:

The trace record does provide target address information for those instructions in which the target address is in some way data dependent.

Examiner assets that the recitation of Mann Col.3 Lines 5-7 is not particularly relevant or relied upon to reject the limitation argued above by the appellant. Mann teaches the limitations presented above, as noted, in previous two pages.

Further Mann clearly states, trace information (provided by TCODE as synchronization marker) cannot be studied without known base (program counter) address (Mann: Col.15 Lines 42-52).

Appellant's argument relating to Col.16 Lines 2-6 cited in claim 1 rejection in office action of March 22, 2005 are noted. This position has been clarified as TCODEs being the synchronization markers (Office Action September 9th, 2005 Pg. 3 Last sentence ¶1) and TSYNCH register holding information for insertion of a synchronizing address information (Mann'270: Col.16, Lines 3-5. Col.16, Lines 25-

29). Further arguments by appellant are moot, as this clarification is acknowledged by appellant on the same page as, "The FINAL REJECTION cited operation of the synchronization register TSYNCH causes generation of the claimed synchronization marker". TSYCH register is a decrementing register, that gets loaded whenever address information is provided and it decrements for every TCODE=0001 entry (Col.16 Lines 40-44).

Examiner thanks the appellant for attempting to clarify the issue in appeal brief on page 8-9 raised in the ADVISORY ACTION, however examiner respectfully disagrees with the appellant argument.

The ADVISORY ACTION by examiner stated:

> "Applicant (Appellant) has argued against the prior art to go around the prior art, however the claims as recited do not contain the elements as argued. For example on Pg. 10 applicant argues There is no teaching in Mann that a full count in the synchronization register TSYNCH causes generation of a new segment address.' (Examiner disagrees - See comments below for remarks on Pg. 9). The limitation relating to 'full count in the TSYNCH register, is not recited in the claim I rejection."

Appellant states:

> This statement misunderstands the Applicant's argument. The FINAL REJECTION cited operation of the synchronization register ***TSYNCH causes generation of the claimed synchronization marker***.
> For that to be true, then following trace outputs that specify addresses must be indicated by "a number of program counter values in the program counter trace stream by which said corresponding program counter value is offset from said synchronization marker in said program counter trace stream." The Applicants urge that the only teaching of indication of such an offset is an offset relative to a segment register. Thus for the operation of the synchronization register TSYNCH to anticipate the claimed synchronization marked, a new segment address must be generated.

Examiner disagrees as Mann clearly states value in the TSYNCH register is loaded in to a counter which counts down and cases insertion (generation) the synchronization marker (the TCODEs) with the base program counter address (*Mann: Col.16 Lines 35-60*):

Referring to FIG. 7, in operation, a counter 701 is set to the value contained in the synchronization register TSYNC 703 whenever a synchronizing trace entry (e.g., containing a branch target address) is generated. Trace control logic 218 determines when a synchronizing trace entry is generated and provides load signal 705 whenever such addresses are generated. This can be summarized as follows. The counter is decremented by one for each TCODE=1, thus providing for a maximum number of consecutive conditional branch instructions.

Thus, counter 701 is reloaded each time a target address is generated or other appropriate TCODE is generated indicating a synchronizing record has been provided. Counter 701 is decremented by one for trace entries not having an address. If the counter reaches zero, an indication 707 is asserted by counter 701 and provided to trace control 218. In response, trace control 218 causes a trace entry to be inserted with a code indicating that it is a synchronization entry (TCODE=0110) and a current program address. The current program address can be, e.g., the most recently retired instruction. In addition, when a synchronizing entry is recorded in the trace cache 200, it can also be provided to trace pins 220 to ensure sufficient availability of synchronizing trace data for full-function ICE equipment. Note that counter circuit 701 may be included in trace control logic 218.

Further, the examiner asserts that a clear argument is presented for the limitation "synchronization register TSYNCH causes generation of the claimed synchronization marker".

As cited in the ADVISORY ACTION this limitation is also not present in the claim 1.

Appellant asserts:

> Since Mann fails to teach this, the operation of the synchronization register TSYNCH fails to anticipate the claimed synchronization marker. The other remarks in the ADVISORY ACTION are similar in that they take issue with a single link in a chain of the Applicants' arguments. The Applicants arguments show that if the Examiner's characterization of Mann is accepted, this leads to contradictions between the teachings of Mann and the recitations of claim 1.
>
> These convolved argument are necessary because the FINAL REJECTION fails to point out with particularly any portion of Mann teaching the recited "expressing said corresponding program counter value as an offset which indicates a number of program counter values in the program counter trace stream by which said corresponding program counter value is offset from said synchronization marker in said program counter trace stream." The Applicants presented this argument to counter a possible interpretation of the Examiner's statements. This would be unnecessary if the Examiner had pointed out where Mann anticipates this "offset" limitation. The Applicants believe the Examiner has not stated where Mann anticipates this limitation because no such teaching exists in Mann. Accordingly, claim 1 is allowable over Mann.

Appellant has *merely restated claim limitation* and has not presented an argument *differentiating prior art from the claim limitation*, instead has attempted to argue limitation not present in the claims.

Regarding Claim 4

The claim limitation being argued is as follows:

> "said identifying step includes detecting occurrences of program counter loads in the data processor; and
> **said identifying step includes counting detected occurrences of program counter loads.**"

Mann '270 teaches identifying step to include counting detected occurrences of program counter load as follows. Each bit in the trace command (TCODE=1) represents a new value of a program counter loaded (caused by branching decision); hence the number of bits used (up to 15 bits in TCPODE =1) in the trace command will indicate the number of program counter loads. The trace command, besides storing the decision for branch, also acts as a counter of how many program counter loads have taken place (Table 6; Col.14 Lines 7-36). The portion cited by the appellant shows the working of the counter however it does not show the TCODE =0001 bits can be used for counting program counter loads. Further, in the response to arguments addressed in the FINAL ACTION, the necessity to know the number of program counter loads is made evident by Mann. Support for which, cited from the FINAL ACTION – response to arguments for claims 4 & 5:

> However, It would still be inherent from Mann's teachings to identify the loading of the program counter (to reset the TSYNCH register value as noted earlier), this time to keep for the purpose of code coverage, execution performance and performance tuning (Col.16 Lines 61-64). Further, the functional need to keep count of the number of loads and offset value is to be able to compress the data transmitted when program counter is referenced against the memory (Specification: Pg.30 Lines 16 onwards). Handling the conditional branches in a single TCODE command and providing complete program counter address, addresses this functional need.

Examiner respectfully disagrees with appellant's arguments and maintains the rejection.

Regarding Claim 5

Appellant argues:

> Claim 5 recites subject matter not anticipated by Mann. Claim 5 recites "*maintaining a running count of a number of program counter loads that have occurred since insertion of the synchronization marker.*" **Such a count requires a counter to be zeroed upon each synchronization marker and incremented upon each detection of a program counter load**. The application teaches this at page 30, lines 13 to 18.

Mann teaches a counter that is decremented from the high-count value stored in the TSYNCH register until a zero is reached (Mann: Col.16 Lines 25-29, 48-53) when the synchronization marker is provided, instead of counter being incremented and then zero when the synchronization marker is provided on a program counter value change.

Appellant's argues:

> Mann teaches that counter 701 is loaded with the synchronization register TSYNC 703 value each time a target address is generated. This occurs upon a program counter load and other conditions. The counter 701 is then decremented on execution of instructions not requiring a new trace address. Thus counter 701 clearly cannot hold the running count recited in claim 5. Accordingly, claim 5 is allowable over Mann.

Although the cited portion is correct, but the interpretation that is based on the broad claim limitation "*maintaining a running count of a number of program counter loads that have occurred since insertion of the synchronization marker*" is that program counter loads are counted since the insertion of the synchronization marker. A narrower limitation however may overcome the cited prior art.

MPEP 2106 states:

> While it is appropriate to use the specification to determine what applicant intends a term to mean, a positive limitation from the specification cannot be read into a claim that does not impose that limitation. A broad interpretation of a claim by Office personnel will reduce the possibility that the claim, when issued, will be interpreted more broadly than is justified or intended. An applicant can always amend a claim during prosecution to better reflect the intended scope of the claim."

Regarding Claim 13

Appellant argues:

> Claim 13 recites subject matter not made obvious by the combination of Sites and Mann. Claim 13 recites "said program counter identifier operable for expressing said corresponding program counter value as an offset which indicates a number of program counter values in the program counter trace stream by which said corresponding program counter value is offset from said synchronization marker in said program counter trace stream." The OFFICE ACTION cites the same portions of Mann as cited against claim 1 as teaching this limitation. Mann likewise fails to teach the synchronization marker or the offset recited in claim 13.

Arguments presented by the appellant are the same as presented for claim 1 and the same response applies to the claim 13 arguments as presented for claim 1 above.

Regarding Claim 15

Arguments presented by the appellant are the same as presented for claim 4 and the same response applies to the claim 15 arguments as presented for claim 4 above.

Regarding Claim 16

Arguments presented by the appellant are the same as presented for claim 5 and the same response applies to the claim 16 arguments as presented for claim 5 above.

Regarding Claim 23 & 24

Arguments presented by the appellant are the same as presented for claim 1 (or claim 13) and the same response applies to the claim 24 & 25 arguments as presented for claim 1 above. Appellant argues that:

> Mann teaches that the trace address values are an offset with a segment base address. Mann includes no teaching that a synchronization marker is the same as the segment base address.

Although, this limitation being argued is not present in the claim 23 or 24, Mann teaches it as synchronization marker providing the segment base address (provided above in Table 6 as TCODE 0011), where the TCODE is synchronization marker inserted in the trace stream.

Further appellant argues:

> Mann teaches providing a new synchronization marker following a taken conditional branch but fails to teach that the trace address values following such a synchronization marker are offset from the synchronization marker value. Instead, Mann teaches that the trace address values are offset from the prior and still unchanged segment base address. The FINAL REJECTION does not allege that Sites or Edwards adds any teaching to Mann to make obvious this limitation. Accordingly, claims 23 and 24 are allowable over the combination of Sites, Mann and Edwards.

Mann in *Col.15 Lines 42-52* teaches TCODE presenting an offset from the known program (counter) address or base address.

> *When executing typical software on a processor-based device 102 according to the disclosed embodiment of the invention,* ***few trace entries contain address values****.* ***Most entries are of the TCODE=0001 format****, in which a single bit indicates the result of a conditional operation. When examining a trace stream, however,* ***data can only be studied in relation to a known program address****. For example, starting with the oldest entry in the trace cache 200, all entries until an address entry are of little use.* ***Algorithm synchronization typically begins from a trace entry providing a target address.***

Appellant's arguments are unpersuasive for the reasons and rejections for claim 23 & 24 presented above.

Regarding Claim 27 & 29 and 28 & 30

No new arguments are presented, other than these claims are allowable based on the their dependency on the argued claims above. Since the rejections for claims being argued above is maintained, claims 27 & 29 and 28 & 30 remain rejected due to their dependency on claim 23 and 24 respectively.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Akash Saxena
Patent Examiner, GAU 2128
(571) 272-8351
Thursday, April 20, 2006

Conferees:

Kamini S. Shah
Supervisory Patent Examiner, GAU 2128

Anthony Knight, SPE 2121
Supervisory Patent Examiner, GAU 2121

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte GARY L. SWOBODA and ROBERT A. MCGOWAN

Application 09/943,599

Corrected
Akash Saxena
11/29/06

ORDER RETURNING UNDOCKETED APPEAL TO EXAMINER

This application was electronically received at the Board of Patent Appeals and Interferences on October 3, 2006. A review of the application has revealed that the application is not ready for docketing as an appeal. Accordingly, the application is herewith being returned to the examiner. The matters requiring attention prior to docketing are identified below.

Upon review of the Examiner's Answer filed on May 3, 2006, there is no indication that an appeal conference has been conducted, since Examiner Akash Saxena did not sign the Examiner's Answer.

See, § 1207.01 of the Manual of Patent Examining Procedure (MPEP) (8th Ed., Rev. 3, August 2005) it states:

Signed

> On the examiner's answer, below the primary examiner's signature, the word "Conferees:" should be included, followed by the typed or printed names of the other two appeal conference participants. These two appeal conference participants must place their initials next to their name. This will make the record clear that an appeal conference has been held. [Emphasis added.]

Accordingly, it is

ORDERED that the application is being electronically returned to the Examiner for taking corrective action regarding the appeal conference, and for such further action as may be appropriate.

BOARD OF PATENT APPEALS
AND INTERFERENCES

By: G. P. Edgell for Dale Shaw

DALE M. SHAW
Deputy Chief Appeals Administrator
(571) 272-9797

GJH

cc: TEXAS INSTRUMENTS INCORPORATED
P.O. BOX 655474, M/S 3999
DALLAS, TX 75265